## REPLACEMENT CLAIMS:

1. (currently amended) A method for implementing interrupts in a data processing system, comprising the steps of:

   providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

   providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

   coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an interrupt request signal which will cause an interrupt to occur in the data processing system; and

   assigning an interrupt prioritization level to storage locations of the first storage device and the second storage device.

   ~~determining priority between two interrupts, a first interrupt being hardware-generated and a second interrupt being software generated, when the two interrupts have a same prioritization level by choosing to service one of the hardware-generated first interrupt or the software-generated second interrupt.~~

2

2. (currently amended) The method of claim 1 further comprising the step of:
   ~~assigning an interrupt prioritization level to storage locations of the first storage~~
   ~~device and the second storage device.~~
   determining priority between two interrupts, a first interrupt being hardware-
   generated and a second interrupt being software-generated, when the two
   interrupts have a same prioritization level by choosing to service one of
   the hardware-generated first interrupt or the software-generated second
   interrupt.

3. (currently amended) The method of claim ~~2~~ 1 further comprising the step of:
   assigning a portion of the plurality of software-generated interrupt signals stored
   in the second storage device to represent interrupts from some interrupt
   sources generating hardware interrupts and having a corresponding
   interrupt prioritization level.

4. (currently amended) The method of claim ~~2~~ 1 further comprising the step of:
   assigning a portion of the plurality of software-generated interrupt signals stored
   in the second storage device to represent interrupts from some interrupt
   sources generating hardware interrupts and having an interrupt
   prioritization level which differs from the interrupt prioritization level of
   the plurality of hardware-generated interrupt sources coupled to the first
   storage device.

5. (currently amended) The method of claim ~~2~~ 1 further comprising the step of:
   changing interrupt servicing from servicing a hardware-generated interrupt and
   switching to servicing a software-generated interrupt of higher
   prioritization before completion of servicing of the hardware-generated
   interrupt occurs.

3

6. (currently amended) The method of claim ~~2~~ 1 further comprising the step of:

changing prioritization level of a predetermined hardware-generated interrupt by providing a software-generated interrupt which represents a corresponding hardware-generated interrupt source for the predetermined hardware-generated interrupt but with a different prioritization level than the predetermined hardware-generated interrupt.

8. (currently amended) A method for implementing interrupts in a data processing system, comprising the steps of:

providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

assigning an interrupt prioritization level to storage locations of the first storage device and the second storage device.

coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an interrupt request signal which will cause an interrupt to occur in the data processing system; and

4

coupling enabling circuitry between the first and second storage devices and the logic circuitry, the enabling circuitry receiving the hardware-generated and software-generated interrupts and determining whether to pass the hardware-generated and software-generated interrupts to the logic circuitry for further processing.

9. (previously amended) A data processing system with interrupt control circuitry, comprising:

a plurality of hardware interrupt sources;

a hardware interrupt storage device having a plurality of inputs, each of the plurality of inputs being coupled by an electrical conductor to one of a plurality of hardware interrupt sources, the hardware interrupt storage device storing hardware-generated interrupts and providing each of the hardware-generated interrupts at a predetermined output terminal;

a software interrupt storage device having a plurality of inputs, each of the plurality of inputs receiving a predetermined one of a plurality of software-generated interrupt signals, at least one of the software-generated interrupt signals corresponding to interrupt servicing of a portion of the data processing system which is not designated as a hardware interrupt source; and

logic circuitry coupled to the hardware interrupt storage device and the software interrupt storage device for providing a data processing system interrupt signal in response to receipt of either hardware-generated interrupts or software-generated interrupts, wherein the logic circuitry determines priority between two interrupts, a first interrupt being hardware-generated and a second interrupt being software-generated, when the two interrupts have a same prioritization level by choosing to service one of the hardware-generated first interrupt or the software-generated second interrupt.

5

10.  (original) The data processing system of claim 9 wherein the hardware interrupt storage device and the software interrupt storage device have an assigned interrupt prioritization level to specific storage locations, the interrupt prioritization level of the hardware interrupt sources being permanently assigned, but assignment of the interrupt prioritization level of interrupt sources associated with the software-generated interrupt signals being variable by software control.

11.  (original) The data processing system of claim 10 wherein a software-generated interrupt signal of higher priority than a currently executing hardware-generated interrupt signal is provided to the logic circuitry prior to completion of an associated hardware interrupt servicing, and the data processing system suspends processing of the hardware interrupt servicing to process an associated software interrupt servicing.

12.  (original) The data processing system of claim 9 further comprising:
     a mask register coupled to the hardware interrupt storage device and the software interrupt storage device for selectively preventing hardware-generated interrupt signals and software-generated interrupt signals from propagating to the logic circuitry.

13.  (original) The data processing system of claim 9 wherein the hardware interrupt storage device and the software interrupt storage device are each implemented as latch circuits.

14.  (original) A method for implementing interrupts in a data processing system, comprising the steps of:
     providing a first storage device having a first plurality of prioritized storage locations representative of priority of interrupt servicing for hardware-interrupt signals stored therein, the first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a physical conductor to a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and storing the hardware

6

interrupts, the first storage device providing one or more hardware-
generated interrupt signals;

providing a second storage device having a second plurality of prioritized storage
locations representative of priority of interrupt servicing for software-
generated interrupt signals stored therein, the second storage device
having one or more inputs, each of the one or more inputs receiving and
storing a predetermined one of a plurality of software-generated interrupt
signals, the predetermined plurality of software-generated interrupt signals
indicating an interrupt from a different source or of a different type than
the hardware interrupts, the second storage device providing one or more
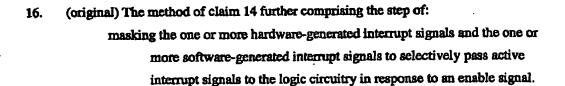software-generated interrupt signals;

executing software with the data processing system to generate a predetermined
software-generated interrupt signal which emulates a predetermined one of
the hardware-generated interrupt sources but with a priority which differs
from the predetermined one of the hardware-generated interrupt sources,
thereby dynamically changing prioritization of servicing of interrupts in
the data processing system; and

coupling logic circuitry to the first storage device and the second storage device
for receiving the one or more hardware-generated interrupt signals and the
one or more software-generated signals, the logic circuitry providing an
interrupt request signal which will cause an interrupt to occur in the data
processing system.

15.     (original) The method of claim 14 further comprising the steps of:

generating the predetermined software-generated interrupt signal which emulates
the predetermined one of the hardware-generated interrupt sources while
another hardware-generated interrupt is being serviced, the predetermined
software-generated interrupt signal having a priority which is higher than
the other hardware-generated interrupt being serviced; and

suspending servicing of the other hardware-generated interrupt being serviced to
begin servicing of the predetermined software-generated interrupt signal.

7

16.    (original) The method of claim 14 further comprising the step of:

      masking the one or more hardware-generated interrupt signals and the one or

          more software-generated interrupt signals to selectively pass active

          interrupt signals to the logic circuitry in response to an enable signal.

17. (currently amended) The method of claim 2 1 further wherein the interrupt prioritization

level is assigned to specific storage locations of the first and second storage device, the

interrupt prioritization level of the plurality of hardware-generated interrupt sources

coupled to the first storage device being permanently assigned, but assignment of the

interrupt prioritization level of interrupt sources associated with the second storage

device being variable by software control.

18.    (cancelled)

19.    (currently amended) The method of claim 18 8 further comprising the step of:

      assigning a portion of the plurality of software-generated interrupt signals stored

          in the second storage device to represent interrupts from some interrupt

          sources generating hardware interrupts and having a corresponding

          interrupt prioritization level.

20.    (currently amended) The method of claim 18 8 further comprising the step of:

      assigning a portion of the plurality of software-generated interrupt signals stored

          in the second storage device to represent interrupts from some interrupt

          sources generating hardware interrupts and having an interrupt

          prioritization level which differs from the interrupt prioritization level of

          the plurality of hardware-generated interrupt sources coupled to the first

          storage device.

21.    (currently amended)  The method of claim 18 8 further comprising the step of:

8

changing prioritization level of a predetermined hardware-generated interrupt by providing a software-generated interrupt which represents a corresponding hardware-generated interrupt source for the predetermined hardware-generated interrupt but with a different prioritization level than the predetermined hardware-generated interrupt.

22.  (previously added)  A method for implementing interrupts in a data processing system, comprising the steps of:

providing a first storage device having a plurality of inputs, each of the plurality of inputs being coupled by a respective physical conductor to one of a plurality of hardware-generated interrupt sources which selectively generate hardware interrupts and selectively storing the hardware interrupts, the first storage device providing one or more hardware-generated interrupt signals;

providing a second storage device having one or more inputs, each of the one or more inputs receiving and storing a predetermined one of a plurality of software-generated interrupt signals, at least some of the predetermined plurality of software-generated interrupt signals indicating an interrupt from a different source or of a different type than the hardware interrupts, the second storage device providing one or more software-generated interrupt signals;

coupling logic circuitry to the first storage device and the second storage device for receiving the one or more hardware-generated interrupt signals and the one or more software-generated signals, the logic circuitry providing an interrupt request signal which will cause an interrupt to occur in the data processing system; and

changing prioritization level of a predetermined hardware-generated interrupt by providing a software-generated interrupt which represents a corresponding hardware-generated interrupt source for the predetermined hardware-generated interrupt but with a different prioritization level than the predetermined hardware-generated interrupt.

9